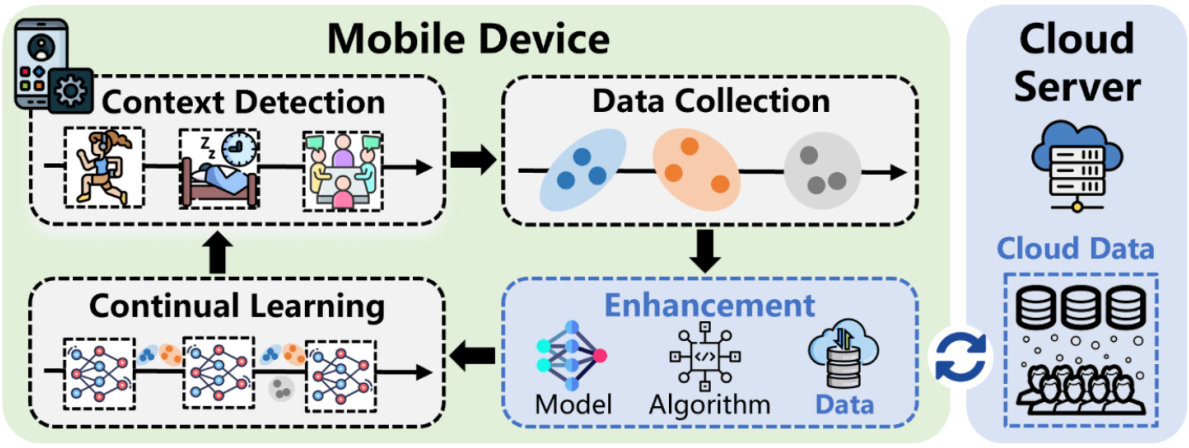
Delta: A Cloud-assisted Data Enrichment Framework for On-Device Continual Learning

关于CL

机器学习(Machine Learning，ML)模型已经成为现代移动应用和服务中不可或缺的组成部分，如Google Smart Lens中的图像标注、Siri中的语音识别、Apple Intelligence中的文本摘要和重写等。

在广泛的移动应用中，用户在日常生活中会遇到动态的情境，并表现出不同的行为，导致移动设备观测和收集的数据分布是非平稳的。

因此，随着新的上下文数据出现，设备上的ML模型期待逐渐进化。这种被称为持续学习(CL)，使得设备上的ML模型能够逐步学习不同情境和行为中的个人用户偏好，从而随着时间的推移变得更加个性化和智能化。



设备上的CL流水线，4个步骤

- 场景检测：当用户经历一个新的上下文时，移动设备可以通过现有的人工或自动方法来检测。
- 数据收集：对于每个新的context，移动设备收集服从新分布的数据样本，作为后续CL过程的训练数据。
- 数据增强：CL之前，需要应用各种增强技术来**减轻数据稀缺带来的严重影响。**
- 持续学习：将**新情境和过去情境的训练数据混合起来**更新设备模型，它是被公认的在不遗忘过去情境知识的情况下从新情境中同化知识的最有效方法之一。

相关工作

云端持续学习（Cloud-Side Continual Learning）：

- 云端持续学习旨在通过对不断变化的数据流进行训练，帮助模型获得新的上下文知识，同时不忘记过去的知识。
- 目前的解决方案包括：
  - 通过惩罚参数变化来稳定已学习的突触变化。
  - 通过扩展和修剪突触连接来创建新的记忆空间。

    - 通过存储和回放过去重要数据来巩固记忆，这些数据的重要性可以通过代表性、多样性或不确定性来度量。
- 研究表明，数据回放方法在模型性能和系统效率之间提供了最佳的平衡。

设备端持续学习（Device-Side Continual Learning）：

- 设备端持续学习聚焦于在资源受限的设备上实现云端持续学习算法，优化硬件资源的利用。
- 包括使用数据量化技术节省存储成本、通过上下文感知的参数稀疏性减少内存开销、使用分层内存管理加速数据加载等方法。
- 然而，很多研究忽视了移动设备上的数据瓶颈（如个人数据稀缺且不可预测），而Delta框架正是为了解决这一问题。

设备端数据增强（On-Device Data Augmentation）：

- 设备端数据增强通过从现有的用户数据中生成多样化数据来提高模型训练性能，包括对视觉图像进行几何和颜色变换、对IMU信号使用物理原理技术、对文本数据进行规则转换和同义词替换等。
- 但数据增强有局限性，因为每种数据类型和任务需要专门设计的增强技术，过程繁琐且低效。
- Delta框架为此提供了一个**通用解决方案**，直接扩展设备端可用的数据，简化了增强过程。

现有工作的局限性

两种典型的应对方法：少样本连续学习（Few-shot CL）和联邦连续学习（Federated CL）。这两种方法分别从模型初始化和训练算法的角度，旨在缓解过拟合和灾难性遗忘问题。

1. **少样本连续学习（Few-shot CL）**：这种方法通过在具有大量数据的常见上下文上预训练模型，以捕捉一般性知识，再通过模型初始化和迁移学习技术将这些知识迁移到新场景中。然而，这种方法在设备端应用中效果不佳，因为未来用户的上下文不可预测且多样化。
    1. FS-KD(知识蒸馏)：通过保持历史数据样本的模型输出不变，将过去的上下文知识蒸馏到新的上下文模型中。
    2. FS-RO（鲁棒优化）：将模型参数约束在所有上下文的训练目标函数的共同平坦极小值内。
    3. FS-PF(参数冻结)：冻结了先前模型训练中的具有高值的重要参数。
2. **联邦连续学习（Federated CL）**：这种方法通过利用云服务器定期汇总分布式设备上训练的本地模型，从而减少单一设备上的过拟合问题，并促进跨设备的知识转移。然而，联邦学习的模型性能和收敛速度对设备的参与率和设备之间的数据异质性非常敏感，这导致了较高的通信开销和不稳定的训练过程，难以在实际应用中稳定运行

关键发现

- 丰富的云端数据资源：云服务器通常拥有大量来自各种渠道的数据，如组织发布的公共数据集（例如ImageNet）、从互联网抓取的开源数据（例如Common Crawl），以及由授权的移动用户贡献的众包数据（例如华为的DonateClient服务、苹果的Learn from this app）
- 用户上下文和行为的相似性：先前的研究表明，尽管不同用户的偏好和行为在不同上下文中有所不同，但它们之间往往存在一定的相似性，而不是完全独特。这意味着云端有一个数据子集，它与设备端的数据具有相似的分布，可以用来提高设备端连续学习（CL）的性能。

主要挑战

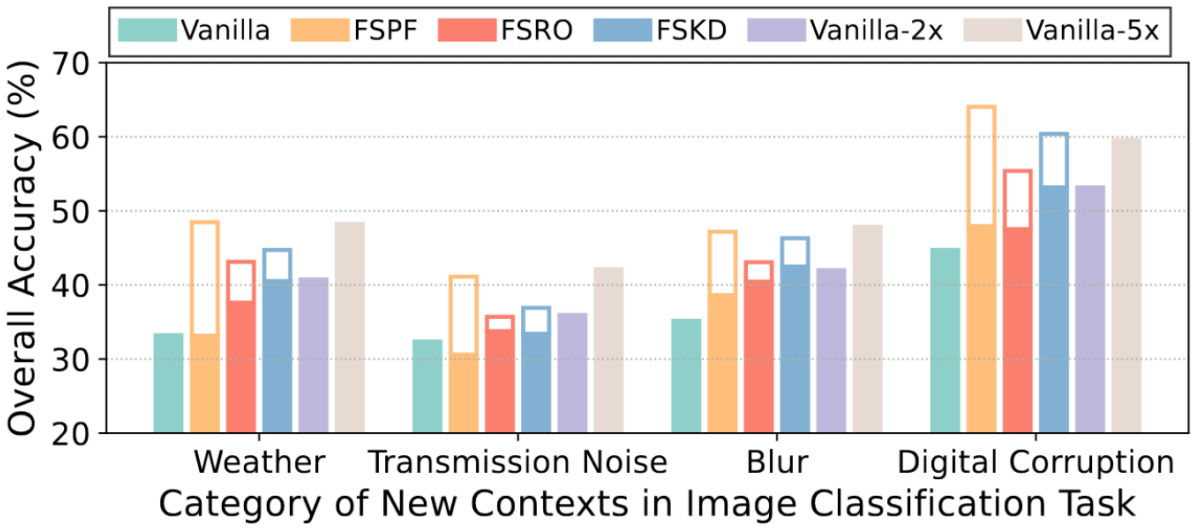
要同时具备私密性、有效性和高效性，很难同时实现。

1. **隐私 vs. 效率**：移动应用中的用户数据通常受到严格的隐私法规（如GDPR）的保护。然而，为了从云端获取最优的数据子集来丰富设备端数据，需要上传用户的原始数据到云端进行精准的相似性比较，或者从云端下载多个数据子集并通过试错方式选择合适的子集。这些方法可能会涉及到隐私泄露的风险，因此，如何在不侵犯用户隐私的前提下高效地进行数据丰富，是一个挑战。
2. **新上下文的有效性 vs. 效率**：云端数据源多样化，随机选择的数据子集可能与设备端数据分布差异较大，从而影响设备端连续学习的效果。然而，为了找到最合适的云端数据子集，**云服务器需要从庞大的云端数据集中评估大量的候选子集**，这会导致计算复杂度和时间开销过高。因此，在保证高效性的同时实现高效的数据丰富效果也是一个挑战。
3. **对过去和新上下文的有效性**：由于移动用户遇到的新上下文的数据分布是动态变化的，如果针对每个新上下文独立进行数据丰富，可能会导致设备端模型在处理过去的上下文时，记忆稳定性下降，因为不同上下文学习过程之间的相互干扰会加剧。此外，**目前缺乏关于新上下文的数据丰富与过去上下文模型表现之间关系的理论分析或见解**，这使得针对新旧上下文的有效数据丰富策略设计变得更加复杂。

贡献

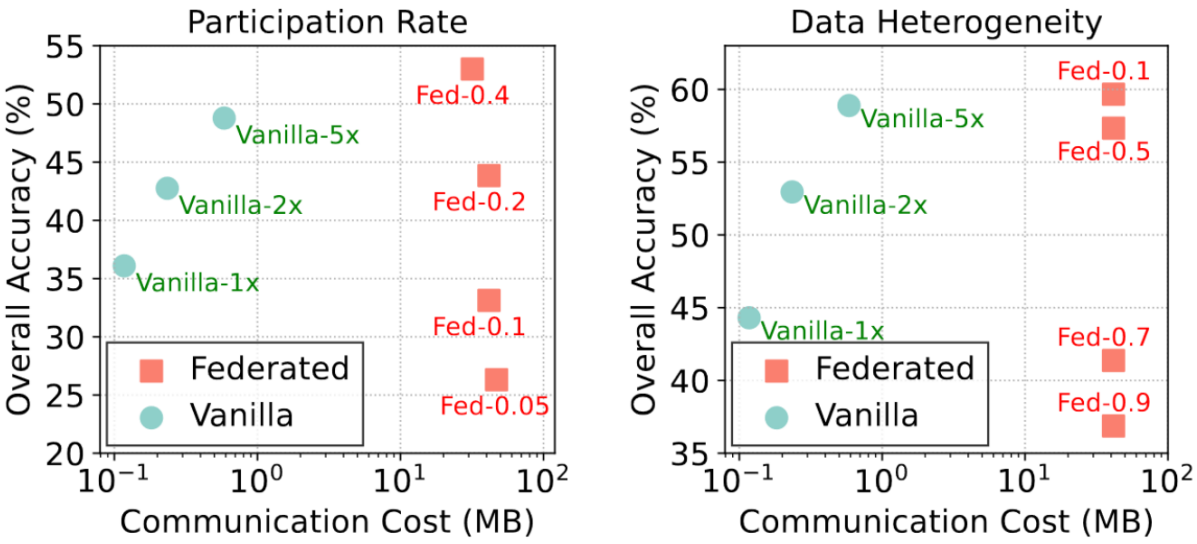
- 首先给出了在设备上CL的数据富集问题的**一般形式化描述**，并分析了在用户隐私和计算效率方面的实际挑战。
- 提出了为云端数据构建一个紧凑的**"目录"数据集**。使得数据富集问题分解为两个子问题，能够由移动设备和云服务器独立解决，不需要交换敏感的原始数据。
- 制定**软数据匹配策略**，精确求解设备侧子问题，并针对云端数据选择提出了一种理论上最优的数据采样方案。
- 从理论上分析了新场景下的数据在所有场景下对模型性能的影响，并从整体角度重新优化了云端数据采样策略。

动机



在没有用户上下文的先验信息的情况下，小样本CL的性能显著下降，模型精度降低范围为8.6 - 15.3 % (FS-PF)，1.9 - 7.9 % (FS-RO)和3.9 - 7.2 % (FS-KD)。

相比之下，简单地将训练数据规模增加到50，就可以超过所有的小样本CL方法，这突出了数据丰富的潜力。



具有不同设备参与率和数据异构程度的联邦CL的通信成本和准确性

Fed - p表示p × 100 %的设备参与率或设备持有来自不同情境的数据。

只有当≥20 %的设备参与每一轮的模型聚合或超过≥30 %的移动用户体验相似的上下文时，联邦CL才能获得优异的性能。

而与联邦CL相比，将数据以合适的分布从云端发送到每个设备可以达到相同的目标精度，且通信成本降低到1 %以下。

文章工作

问题的形式化建模分析

D_{de}^{t}：每个情景t = 1，..，T的底层数据分布。

$\hat{D}_{de}^{t}$：设备收集的用于训练on - device模型的经验数据集。

S^{t}：从云端检索的相似数据集，用于增强设备的$\hat{D}_{de}^{t}$。

D_{cl}：云端数据集。

两个数据集D1和D2关于模型θ的训练过程的相似性：

式（1）

$$Sim(\mathcal{D}_1, \mathcal{D}_2|\theta) \triangleq -\max_{\|\theta'-\theta\|\leq\epsilon} \left\|\nabla L(\mathcal{D}_1, \theta') - \nabla L(\mathcal{D}_2, \theta')\right\|$$

$L(D,\theta) = E_{(x,y)\in D}[l(x,y,\theta)]$表示模型θ在数据集D上的期望损失。

两个数据集D1和D2之间的高度相似性意味着它们在多个步骤中更新模型参数的性能相当，从而对设备上模型训练产生类似的影响。

对于每个context，云服务需要选择**最相似**的数据子集$S^{t,*}$，更新当前的设备模型θ^{t-1}.

$S^{t,*}$和D_{de}^{t}根据上式，应该表现出很高的相似性。

式（2）

$$\begin{aligned} \mathcal{S}^{t,*} &= \underset{\mathcal{S}^t \subseteq \mathcal{D}_{cl}, |\mathcal{S}^t| \leq B}{\arg\max} \; Sim(\mathcal{S}^t, \mathcal{D}_{de}^t \mid \theta^{t-1}) \\ &\approx \underset{\mathcal{S}^t \subseteq \mathcal{D}_{cl}, |\mathcal{S}^t| \leq B}{\arg\max} \; Sim(\mathcal{S}^t, \widehat{\mathcal{D}}_{de}^t \mid \theta^{t-1}), \end{aligned}$$

B表示所选数据子集的最大允许大小。

该式子面临两个挑战

- 隐私担忧：设备需要上传当前的θ^{t-1}和原生数据$\hat{D_{de}^t}$，对用户隐私构成严重侵犯。
- 云服务器计算负担：云服务器需要为每个可能的数据子集计算相似度，导致指数计算复杂。

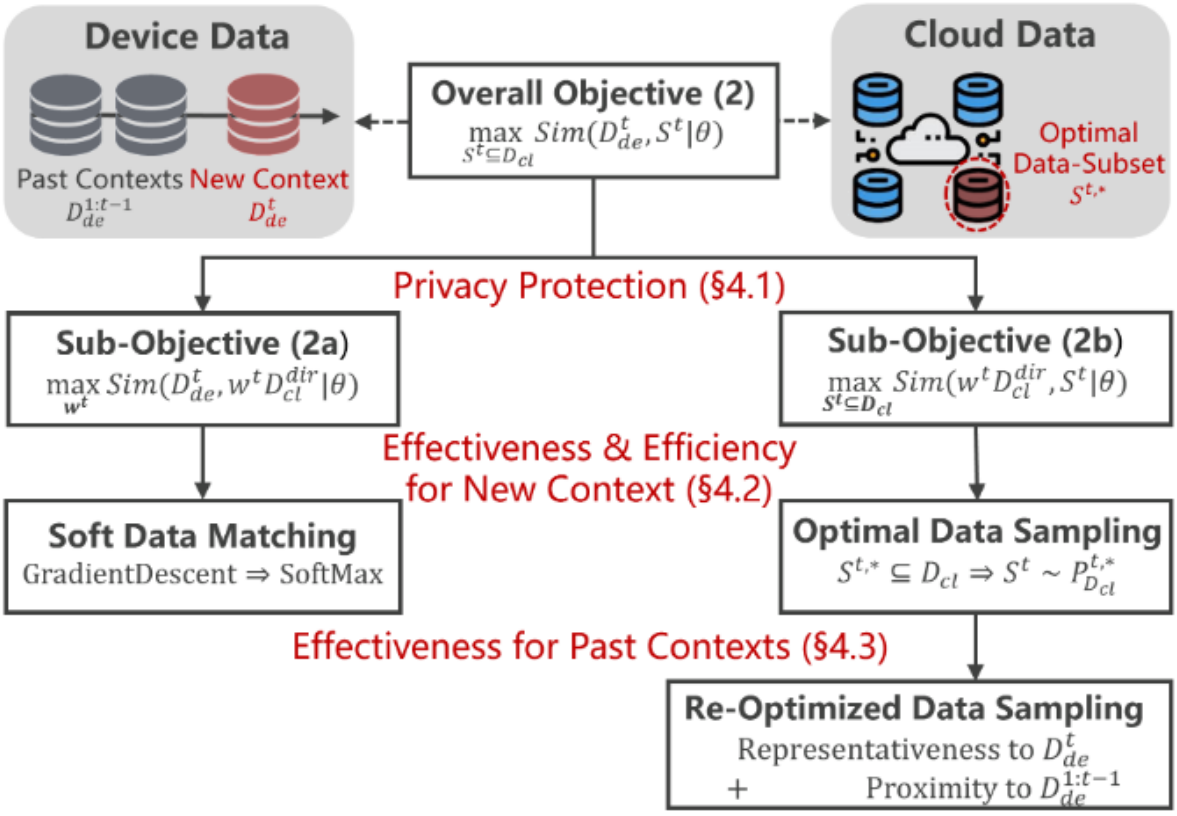
架构设计

目录数据集构造

Delta构造了一个由少量数据样本组成的紧凑**目录数据集**D_{cl}^{dir}，来表示广泛的云端数据集。

式（2）可以被拆解为两个子目标问题：

- 子目标(2a)：端侧数据集D_{de}^{t}和加权目录数据集$w^t * D_{cl}^{dir}$的相似性，其中，每个数据样本$(\overline{x_c}, \overline{y_c}) \in D_{cl}^{dir}$被赋予一个权重$w_c^t$。权重向量$w^t$**是待优化**的变量。
- 子目标(2b)：加权目录数据集$w^t * D_{cl}^{dir}$与云端数据子集S^t的相似性，S^t**是待优化**的变量。



这种分解为啥可行?

$$\underbrace{\max_{\mathcal{S}^t \subseteq \mathcal{D}_{cl}} Sim(\mathcal{D}_{de}^{t}, \mathcal{S}^t|\theta^{t-1})}_{\text{original objective}} \geq \underbrace{\max_{w^t} Sim(\mathcal{D}_{de}^{t}, w^t \mathcal{D}_{cl}^{dir}|\theta^{t-1})}_{\text{sub-objective (2a) for optimal weight}} + \underbrace{\max_{\mathcal{S}^t \subseteq \mathcal{D}_{cl}} Sim(\mathcal{S}^t, w^{t,*} \mathcal{D}_{cl}^{dir}|\theta^{t-1})}_{\text{sub-objective (2b) for optimal subset}},$$

原目标函数的最优值是两个子目标函数(2a)和(2b)的最优值之和的上界。原始目标函数与子目标函数之间的实际差距由**云端目录数据集的代表性**决定。

通过交换非敏感信息，它们可以由端设备和云端独立依次优化。

- 移动设备通过计算目录数据集D_{cl}^{dir}的**最优权重**$w^{t,*}$表示设备端数据分布D_{de}^{t}，优化子目标(2a)
- 云服务搜索最优的云端数据子集$S^{t,*}$，与加权目录数据集对齐（权重$w^{t,*}$由端侧设备上传）
- 端云的通信包括云端目录数据集D_{cl}^{dir}和端侧优化的权重$w^{t,*}$，**不包括任何用户原本数据。**

以上的问题拆解和独立优化，最大依赖于一个能够**准确表示**云端公共数据集的目录数据集。

由于云端数据来源多样、分布广泛、量纲各异，直接对原始数据样本进行聚类，以簇心作为目录元素，可能会无法完全捕捉数据对模型训练的影响。

作者利用设备上模型训练的典型范式，在**云上预训练**一个**特征提取器**ϕ（在设备模型训练过程中保持不变），并在**端侧数据上**训练一个**分类器**ψ。提出**基于特征提取器输出**$\phi(x)$对数据样本$(x, y) \in D_{cl}$进行聚类，并选择聚类中心作为目录数据集的元素。$(\overline{x_c}, \overline{y_c})$是云端数据簇心。

- 特征作为模型的中间输出具有一致的维度，与原始输入相比，与模型训练更相关。
- 大部分特征可以从特征提取器的预训练过程中获得，产生的额外成本最小。

新场景下的数据富集

软数据匹配

要解决2a，需要确定最优权重$w^{t,*}$，使目录数据集与端侧数据分布对齐。然而，端侧数据分布通常由移动设备存储的稀疏经验数据集$\hat{D}^t_{de}$来近似，导致传统的梯度下降算法收敛到**局部最优**，同时，得出的权重变得**过拟合**于有限的经验数据集。

作者对权重w^t赋予了新的物理含义：将每个w^t_c解释为与云端聚类质心$(\overline{x_c}, \overline{y_c}) \in D^{dir}_{cl}$具有**较高相似度**的设备端数据的**分数**。

对每个**端侧数据样本**$(x, y) \in \hat{D}^t_{de}$，将计算它**与所有簇心**的相似度。

- 硬匹配：将最相似的权重逐级递增。

$$c^* = \arg\max_c Sim\big((x, y), (\bar{x}_c, \bar{y}_c) \mid \theta^{t-1}\big)$$

$$w^t_{c^*} \leftarrow w^t_{c^*} + 1.$$

- 软匹配：然而，设备上的数据样本可以表现出与多个云端聚类中心的高度相似性，硬匹配函数**argmax无法捕捉这种关系**。所以作者使用了软匹配函数，利用softmax函数，把端侧数据对每个簇心数据的相似度做归一化，添加到权重。

$$\forall c, w^t_c \leftarrow w^t_c + Softmax\left(\frac{Sim\big((x, y), (\bar{x}_c, \bar{y}_c) \mid \theta^{t-1}\big)}{\tau}\right)$$

其中，τ是温度超参数，控制相似程度不同的簇的权重增量。

最优数据采样

为了减少云端的计算开销，将"硬"的数据选择过程转化为"软"的数据采样过程。从选择一个精确的数据子集$S^{t,*}$，变成计算一个采样策略$P^{t,*}_{D_{cl}}$，对于子问题2b，采样的数据子集是最优的。

$$\max_{\mathcal{S}^t \subseteq \mathcal{D}_{cl}} Sim(\mathcal{S}^t, w^t \mathcal{D}^{dir}_{cl} | \theta^{t-1})$$

$$\Rightarrow \max_{P^t_{\mathcal{D}_{cl}}} \mathbb{E}_{\mathcal{S}^t \sim P^t_{\mathcal{D}_{cl}}} \left[Sim(\mathcal{S}^t, w^t \mathcal{D}^{dir}_{cl} | \theta^{t-1})\right]$$

云服务能够通过数据采样策略直接识别合适的数据子集，该策略可以以恒定的时间复杂度进行计算。

云端数据采样方案的具体操作：

- 簇间大小分配。需要为不同的数据簇分配不同的采样大小，以满足2b。定义每个簇$D_{cl,c}$的**最优采样大小**为$|S_c^{t,*}|$，由**目录权重**w_c^t和**簇内特征分布的离散程度**$E_x||\phi(x)-\phi(\overline{x})||$决定。

$$|\mathcal{S}_c^{t,*}| \propto w_c^t \cdot \mathbb{E}_{(x,y)\in\mathcal{D}_{cl,c}} \left\|\phi(x)-\phi(\bar{x}_c)\right\|$$

- 簇内数据采样。在每个云端数据簇$D_{cl,c}$内，每个数据样本(x,y)的**最优采样概率**正比于该数据样本与聚类中心$(\overline{x_c},\overline{y_c})$之间的特征距离。

$$P_{\mathcal{D}_{cl,c}}^{t,*}(x,y) = \frac{\left\|\phi(x)-\phi(\bar{x}_c)\right\|}{\sum_{(x',y')\in\mathcal{D}_{cl,c}} \left\|\phi(x')-\phi(\bar{x}_c)\right\|}$$

  这种采样策略偏向于距离聚类中心较远的数据样本。

为啥采集策略的数据集能代替精确选择的数据集?

Theorem 2：加权目录数据集$w^t * D_{cl}^{dir}$与根据采样方案$P_{D_{cl}}^t$选取的数据子集S^t之间的期望相似度的下界为：

$$\mathbb{E}_{\mathcal{S}^t\sim P_{\mathcal{D}_{cl}}^t}\left[Sim(\mathcal{S}^t, w^t\mathcal{D}_{cl}^{dir} \mid \theta^{t-1})\right]$$

$$\geq -\mathbb{E}_{\mathcal{S}^t\sim P_{\mathcal{D}_{cl}}^t} L_\psi \left\|\mathbb{E}_{(x,y)\in\mathcal{S}^t}\left[\phi(x)\right] - \sum_c w_c^t\phi(\bar{x}_c)\right\|$$

推导：

1.相似性的近似表达：

$$\mathrm{Sim}(S^t, w^t D_{\mathrm{dir}}^{\mathrm{cl}}) \propto -\left\|\mathbb{E}_{(x,y)\in S^t}\phi(x) - \sum_c w_c^t\phi(\bar{x}_c)\right\|$$

前项表示采样数据子集 S^t 的特征均值，后项表示加权目录数据的加权中心特征。

2.引入 Lipschitz 连续性：由于

$$\|\psi(\phi(x_1)) - \psi(\phi(x_2))\| \leq L_\psi\|\phi(x_1)-\phi(x_2)\|$$

可得：

$$\mathrm{Sim}(S^t, w^t D_{\mathrm{dir}}^{\mathrm{cl}}) \geq -L_\psi\left\|\mathbb{E}_{(x,y)\in S^t}\phi(x) - \sum_c w_c^t\phi(\bar{x}_c)\right\|$$

3.取采样策略下的期望，即得Theorem 2 的最终形式。

为啥采用这样的采样策略?

Lemma1: 采样数据子集 S^t 和加权目录数据集$w^t * D_{cl}^{dir}$之间的期望相似度由每个簇c的采样大小$|S_c^t|$和簇内数据采样概率$P_{D_{cl,c}}^t(x,y)$决定

$$\min_{P_{\mathcal{D}_{cl}}^t} \mathbb{E}_{\mathcal{S}^t \sim P_{\mathcal{D}_{cl}}^t} \left\| \mathbb{E}_{(x,y)\in\mathcal{S}^t}\left[\phi(x)\right] - \sum_c w_c^t \phi(\bar{x}_c) \right\|$$

$$= \min_{|\mathcal{S}_c^t|, P_{\mathcal{D}_{cl,c}}^t} \sum_c \left(\frac{(w_c^t)^2}{|\mathcal{S}_c^t|} \cdot \sum_{(x,y)\in\mathcal{D}_{cl,c}} \frac{\left\|\phi(x) - \phi(\bar{x})\right\|^2}{|\mathcal{D}_{cl,c}|^2 \cdot P_{\mathcal{D}_{cl,c}}^t(x,y)} \right)$$

推导:

1. S^t 的特征期望按照c的贡献分解:

$$\mathbb{E}_{(x,y)\in S^t}\phi(x) = \sum_c \frac{|S_c^t|}{|S^t|} \cdot \mathbb{E}_{(x,y)\in S_c^t}\phi(x)$$

2. S^t 与加权目录的特征值差分解到各个集群c的形式为:

$$\left\| \mathbb{E}_{(x,y)\in S^t}\phi(x) - \sum_c w_c^t \phi(\bar{x}_c) \right\|^2 \approx \sum_c \frac{(w_c^t)^2}{|S_c^t|} \mathbb{E}_{(x,y)\in D_{\mathrm{cl},c}} \left\|\phi(x) - \phi(\bar{x}_c)\right\|^2$$

这里出现了平方，是因为统计学中的方差加权性质，误差的权重贡献是 w_c^t 的平方项。由于每个集群内的特征距离是独立的，整体误差可以看作各集群误差的加权和。

3. 对于集群内的特征差异，引入采样概率$P_{D_{cl,c}}^t(x,y)$:

$$\mathbb{E}_{S_c^t}\left[\left\|\phi(x) - \phi(\bar{x}_c)\right\|^2\right] \approx \sum_{(x,y)\in D_{\mathrm{cl},c}} \frac{1}{P_t^{D_{\mathrm{cl},c}}(x,y)} \left\|\phi(x) - \phi(\bar{x}_c)\right\|^2$$

在概率论中，如果一个数据点 (x,y) 被采样的概率是 $P_{D_{cl,c}}^t(x,y)$，那么它在总体期望中的贡献需要除以这个概率。

直观上，稀有的数据点（采样概率小）被选中的概率低，但一旦被选中，它对误差的贡献需要放大，才能反映其真实的误差大小。

由上式，可以分析出

$$\mathcal{P}_{\mathcal{D}_{cl,c}}^{t,*}(x,y) \quad \propto \quad \left\|\phi(x) - \phi(\bar{x}_c)\right\|, \quad \forall (x,y) \in \mathcal{D}_{cl,c}$$

因为对于特征误差较大的数据点，应该频繁采样，以便更好地代表集群的特征分布。这使得总体误差项被有效地最小化。

4. 不同集群的权重 w_c^t 决定了它们对整体误差的贡献程度，而采样大小 $|S_c^t|$ 决定了我们能够多大程度上减少这一贡献。

要最小化S^t 与加权目录的特征值距离，$|S_c^{t,*}|$ 需要正比于:

$$|\mathcal{S}_c^{t,*}| \quad \propto \quad w_c^t \cdot \mathbb{E}_{(x,y)\in\mathcal{D}_{cl,c}} \left\| \phi(x) - \phi(\bar{x}_c) \right\|$$

对所有场景的数据富集

之前的方法，已经确保了在每个新上下文的私有、高效和有效的数据丰富过程，但臭名昭著的**灾难性遗忘**(即较差的记忆稳定性)问题也加剧了：

- 随着模型参数θ不断适应富集的数据$\{S^i\}_{i=1}^t$，每个过去上下文i的富集数据S^i与底层分布D_{de}^i之间的相似性逐渐降低，阻碍了保留过去的知识。
- 仅仅针对新情境独立地丰富数据，会加剧新、旧情境的模型训练过程之间的相互干扰。

理论分析

通过模型在**所有场景**下的平均损失来量化的总CL性能，主要由三项因素决定：

- 新场景的代表性：由富集数据集S^t与新场景的数据分布D_{de}^t的特征距离量化。
- 过去情境的邻近性：由富集数据集S^t与所有过去的数据分布$\{D_{de}^i\}_{i=1}^{t-1}$的特征距离测量。
- 跨情境的异构性：一个**固定项**，由移动用户所遇到的新情境与过去情境之间的异构性决定。

实际实现

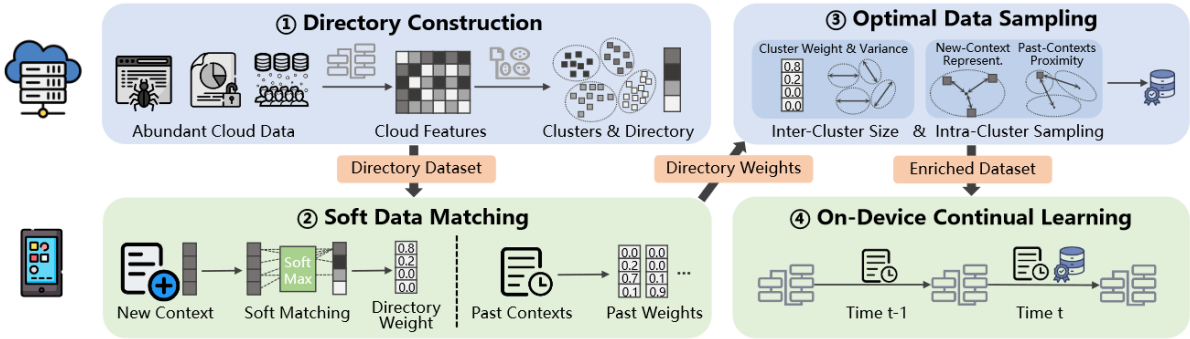
重优化的云端数据采样策略解析表达式：

$$P_{\mathcal{D}_{cl,c}}^{t,*}(x,y) \propto \left\| \phi(x) - \phi(\bar{x}_c) \right\| + \alpha \left\| \phi(x) - \frac{\sum_{i=1}^{t-1} \phi(w^{i,*}\mathcal{D}_{cloud}^{dir})}{t-1} \right\|$$

对于簇内数据采样，每个数据样本的最有采样概率与**新场景代表性**和**过去情境邻近性**的**权重和**成正比。

α是一个由设备确定的超参数，平衡模型在新上下文和过去上下文上的性能。

整体架构



1. 目录构建：首先，云服务器利用预训练的特征提取器从多样化的数据集中**提取特征**，并进行**数据聚类**以**构建目录数据集**。然后，随着模型的部署，目录数据集被**分发到移动设备**中。
2. 软数据匹配：对于每一个即将到来的新上下文t，移动设备通过软数据匹配策略来解决子问题(2a)，并将**新场景**和**过去场景**的最优目录权重**上传至云端**。
3. 最优数据采样：云服务器收到目录权重后，计算最优数据采样方案的解析表达式，其中包括**簇间大小分配**和**簇内数据采样**。方案采样出最优的数据子集，并传输回移动设备。
4. 设备上持续学习：移动设备利用新情境和过去情境的富集数据集进行持续学习。

实验设置

任务，数据集，模型介绍

在4个具有不同数据模态、模型结构和用户情境类别的典型**移动计算任务上**对Delta进行了评估。

Task	Modality	Context Category	Dataset	Model(#params)
IC	Image	Object (O), Weather (W), Noise (N), Blur (B), Digital Corruption (D)	Cifar10-C	ResNet18(11.2M)
HAR	IMU	Activity (A), Physical Condition (P), Device Placement (D)	HHAR, UCI, Motion, Shoaib	DCNN(17.3K)
AR	Audio	User Command (C), Tone (T), Environmental Noise (N)	Google Speech	VGG11(9.75M)
TC	Text	Article Topic (T), Language (L)	XGLUE	BERT(0.178B)

图像分类（Image Classification, IC）：

- 使用了Cifar10-C数据集，该数据集包含了约750,000张图片，分类为10种物体，并且分为**四个背景类别**：天气、噪声、模糊和数字腐蚀。
- 每个背景类别下，数据集被处理成**5个子集**，每个子集包含2个新物体和1个新背景。
- 训练模型：ResNet18，用于进行10类物体的图像分类。

人体活动识别（Human Activity Recognition, HAR）：

- 使用了四个公开数据集：HHAR、UCI、MotionSense和Shoaib，这些数据集包含了来自73个用户的数据，用户进行6种基本活动（静止、行走、上下楼梯、慢跑、骑车），并且数据被收集于5种设备位置（口袋、腰带、手臂、手腕、腰部）。
- 每个场景类别下，数据集被处理成**6个子集**，每个子集包含1个新活动和新的背景。
- 训练模型：基于轻量级CNN的模型DCNN，用于进行6类活动识别。

音频识别（Audio Recognition, AR）：

- 使用了Google Speech Command数据集，该数据集包含了100,000个音频文件，记录了20个命令的声音，来自2000多位用户，声音的语调和环境条件不同。
- 每个场景类别下，数据集被处理成**5个子集**，每个子集包含4个新命令和1个新背景。
- 训练模型：深度神经网络VGG-11，用于音频命令识别。

文本分类（Text Classification, TC）：

- 使用了XGLUE基准中的NC语料库，该语料库包含了50,000篇文章，涉及10个话题，并且有5种语言（德语、英语、西班牙语、法语、俄语）。
- 每个场景类别下，数据集被处理成**5个子集**，每个子集包含2个新话题和1个新背景。
- 训练模型：基于Transformer的BERT模型，用于进行10类文本分类任务。

将设备上上下文的总数标准化为大约5个，确保评估的一致性。

数据配置

数据收集与分配：

- 对于每个任务，收集50%的用户数据（或者对于IC和TC任务，随机选择50%的样本）来形成**云端公共数据集**。剩余的数据则用于模拟在不同背景下的**设备端实测数据**。

云端数据：

- 从不同用户和不同背景中**混合**数据样本，以反映典型的场景，即每个原始数据样本的具体**背景未知**。也就是说，云端的数据包含了多样化的背景和用户。

设备端数据：

- 在每个情景下，选择个label中的5个样本作为**设备端实测数据**，用于模型的微调。这与统计数据表明的平均每个欧洲人每天拍摄约4.9张照片并多次使用Siri一致。
- 剩余的数据样本则作为**测试数据**，用于每个背景的测试。

Delta算法设置：

- 对于设备端软匹配，温度τ设定为0.1。这个参数决定了软匹配的灵敏度。
- **云端数据聚类**：云端的数据聚类数量为20×类别数（IC/HAR/AR/TC任务分别为200、120、400、200）。
- 超参数α设置为1.0，用来平衡云端数据采样对新旧背景的影响。
- **通信预算**：默认的通信预算为每个新背景**25个样本/类**。该预算决定了每个新背景下与设备端模型交互时的**数据量**。

Baselines

- Few-shot CL
- Federated CL
- 随机数据富集：选择一个随机的云端数据子集，以丰富设备侧的经验数据。

Metrics

- 总性能：最终模型在所有上下文中的平均推理准确性。
- 学习可塑性：学习过程中，每个新上下文的最高的准确率的平均值。
- 记忆稳定性：每个上下文的最终准确率与其最大准确率之间的比率的平均值。
- 系统开销：设备侧和云侧的计算延迟、通信成本、内存占用和能耗。

实验结果

Tasks	Context Category	Vanilla CL	Few-Shot CL			Federated CL			Data Enrichment		ΔAcc.	ΔComm.
			FS-KD	FS-RO	FS-PF	Fed-0.1	Fed-0.2	Fed-0.4	Random	Delta		
IC	O+W	32.7±1.49	41.7±1.78	39.2±2.13	36.9±2.87	31.8±0.24	46.4±1.65	55.1±0.42	42.5±2.42	57.7±0.54	16.0% ↑	93.7% ↓
	O+N	31.3±1.74	36.2±2.34	35.5±1.65	32.3±1.25	31.1±0.04	40.4±0.51	45.0±0.12	35.8±1.00	50.9±1.66	14.8% ↑	93.5% ↓
	O+B	35.6±0.94	43.7±1.12	40.6±0.24	39.2±0.06	32.6±0.16	39.6±0.24	50.1±0.31	39.9±1.69	57.7±0.98	14.0% ↑	91.1% ↓
	O+D	45.0±2.57	55.1±1.17	51.5±2.66	52.2±3.10	36.9±0.04	49.0±0.51	61.7±0.34	53.7±2.24	72.3±2.27	17.1% ↑	92.2% ↓
	O+W+N+B+D	77.3±0.49	81.2±1.53	80.4±0.81	75.3±0.41	30.0±0.05	39.8±0.71	50.8±0.41	47.8±6.64	94.8±2.74	13.6% ↑	95.3% ↓
HAR	A	52.4±3.67	55.0±3.93	52.9±2.55	48.3±2.69	54.0±0.64	60.0±0.21	61.3±0.55	58.4±0.35	69.3±1.96	14.3% ↑	99.6% ↓
	A+P	51.2±4.53	53.3±3.20	50.1±3.52	49.4±2.95	60.5±1.28	61.1±1.89	63.1±0.85	58.5±0.75	66.6±1.78	13.3% ↑	99.8% ↓
	A+P+D	81.0±4.75	80.3±2.35	78.7±4.37	71.0±4.27	62.2±3.58	66.8±3.97	70.1±4.28	61.1±3.25	90.3±5.09	10.0% ↑	99.7% ↓
AR	C	93.6±0.16	93.5±0.07	92.9±0.65	94.2±0.28	88.1±1.65	88.3±0.83	88.5±1.78	90.4±0.19	94.3±0.17	0.2% ↑	99.9% ↓
	C+T	89.0±0.41	89.4±0.57	89.4±0.38	90.3±0.79	86.5±0.24	88.5±0.62	88.7±0.25	90.3±0.26	91.1±1.17	0.8% ↑	99.9% ↓
	C+T+N	84.7±0.64	84.8±1.52	86.2±0.79	86.9±0.40	87.5±0.54	87.7±0.31	88.0±0.61	88.5±1.45	89.2±1.60	2.3% ↑	99.9% ↓
TC	T	73.2±2.15	73.5±1.35	75.7±4.07	73.3±2.56	79.6±0.37	79.6±0.19	79.8±0.14	73.9±2.69	83.1±2.26	7.3% ↑	99.8% ↓
	T+L	77.7±3.19	82.2±0.29	80.1±3.02	80.0±1.89	84.3±0.14	84.4±0.18	84.7±0.09	79.7±2.21	86.2±2.16	4.0% ↑	99.4% ↓

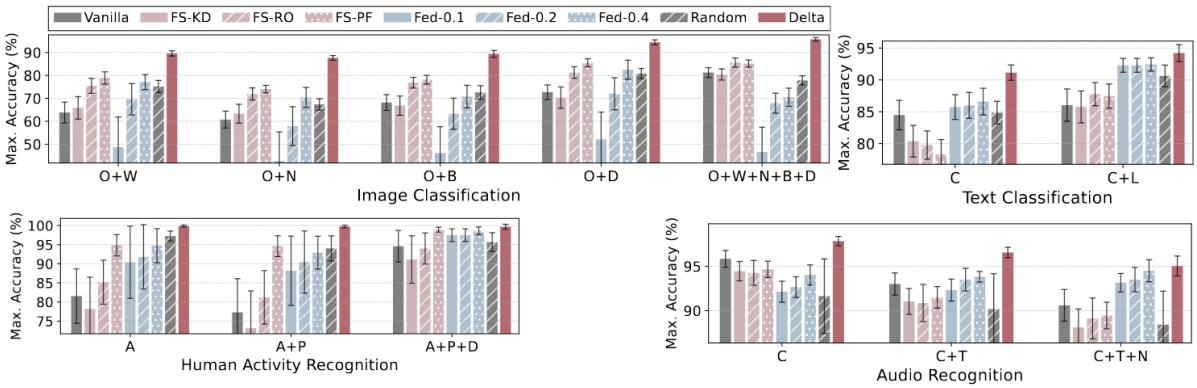
总体CL性能总结

- 与表现最好的Few-shot CL方法相比，Delta方法取得了显著的改进。

在IC上的准确率提高了13 - 16 %，在HAR上的准确率提高了10 - 14 %，在AR上的准确率提高了0.2 - 2.5 %，在TC上的准确率提高了4 - 7.3 %。

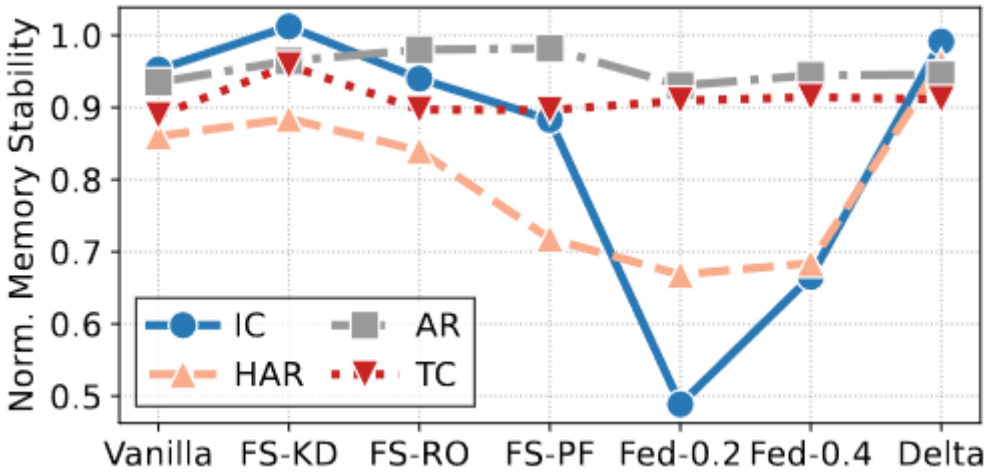
值得注意的是，Delta对AR任务的改进微乎其微，因为其跨情境(即声调和背景噪声)的数据异质性较低，原本的CL能表现良好。

- 与联邦协作学习相比，Delta在所有设置下均获得了最高的整体性能，并将总通信成本降低了91 % ~ 99 %，表明其在提高协作学习性能方面具有更高的有效性和效率。
- 对于大多数任务(IC、HAR和TC)，所有方法都倾向于在混合类别的上下文中表现更好。可能的原因是，不同情境类别的数据样本表现出更大的分布发散性，使得设备上模型**更容易学习到决策边界。**

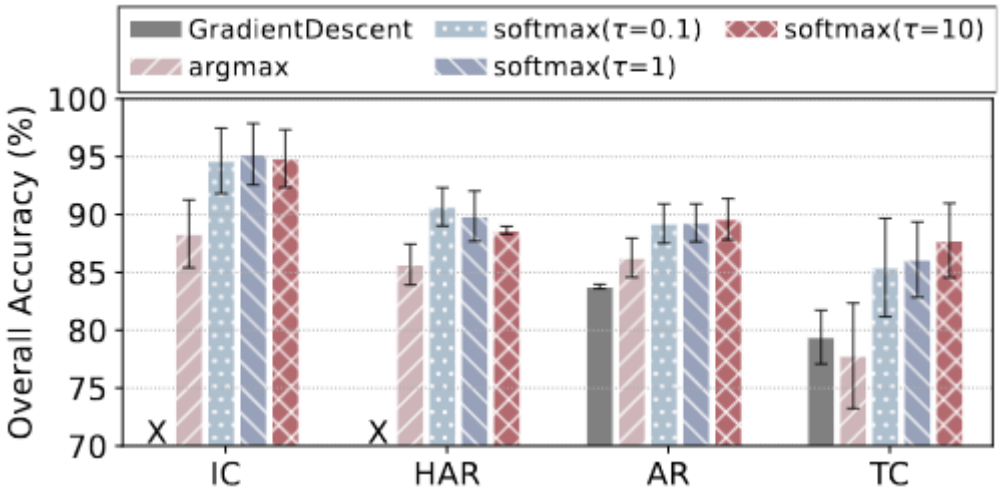


学习可塑性的比较

Delta在IC和HAR任务中对新情境的准确率分别达到90 %和100 %，而在其他两个任务上的准确率波动小于3 %，表现出对多样化新情境的高度鲁棒性和适用性。



Delta对于过去的情境可以保持90 %以上的相对表现，归因于在云端数据采样过程中考虑了新上下文的丰富对所有上下文整体性能的影响。



不同软数据匹配策略的评估

包括梯度下降(GD)、硬匹配(argmax)和变温度τ的软匹配(softmax).

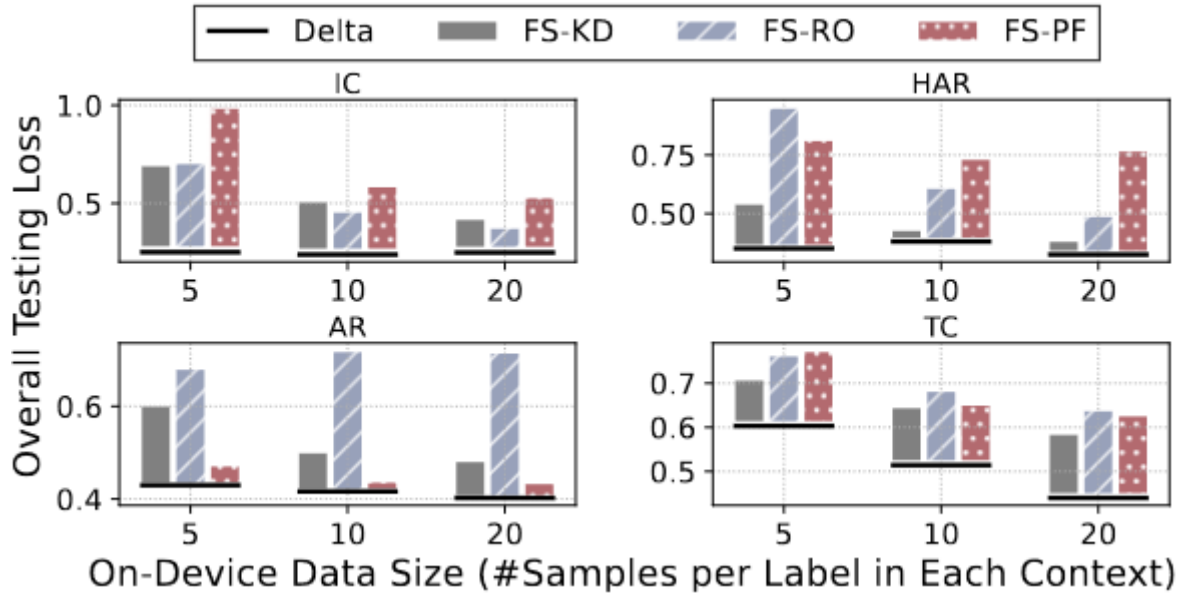
GD在大多数任务上表现不佳，而softmax则始终如一地优于argmax.

原因：

1) GD容易陷入局部最优，导致目录权重过拟合；

2) argmax没有利用设备端样本与多个云端簇之间的相似性，这在云端数据精细聚类时是必不可少的。

此外，由于特征分布的不同，最优的τ因任务而异，我们设定τ= 1.0以获得稳定的性能。



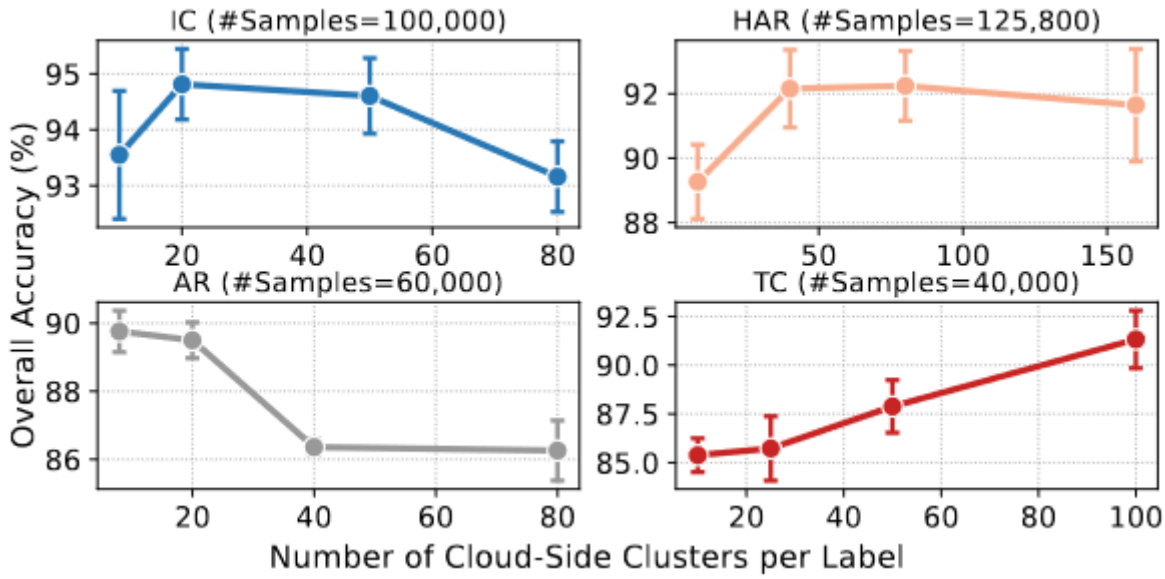
设备端对数据量的敏感性分析

Delta表现出相对较高的鲁棒性，这归因于：

1)通过我们的软匹配策略有效地解决了设备侧数据稀缺的设备侧子问题；

2)丰富的云端丰富数据相对于额外的设备侧用户数据带来了实质性的性能提升。

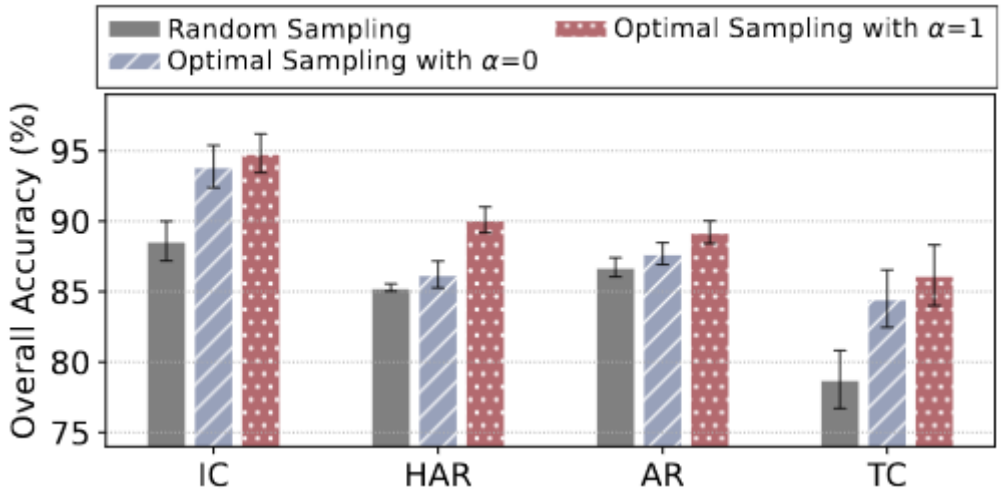
此外，基线对设备上的数据量表现出更大的敏感性，强调了设备上数据丰富的关键作用，并进一步推动了我们的工作。



将云端数据采样方案替换为随机抽样，以隔离目录数据集的影响。

通过使目录数据集更具有代表性，并将云端子目标(2b)与总目标更紧密地对齐，集群数量的略微增加可以提高Delta的性能。然而，过大的聚类数目会导致大量相似的聚类，从而导致选择冗余数据进行富集。

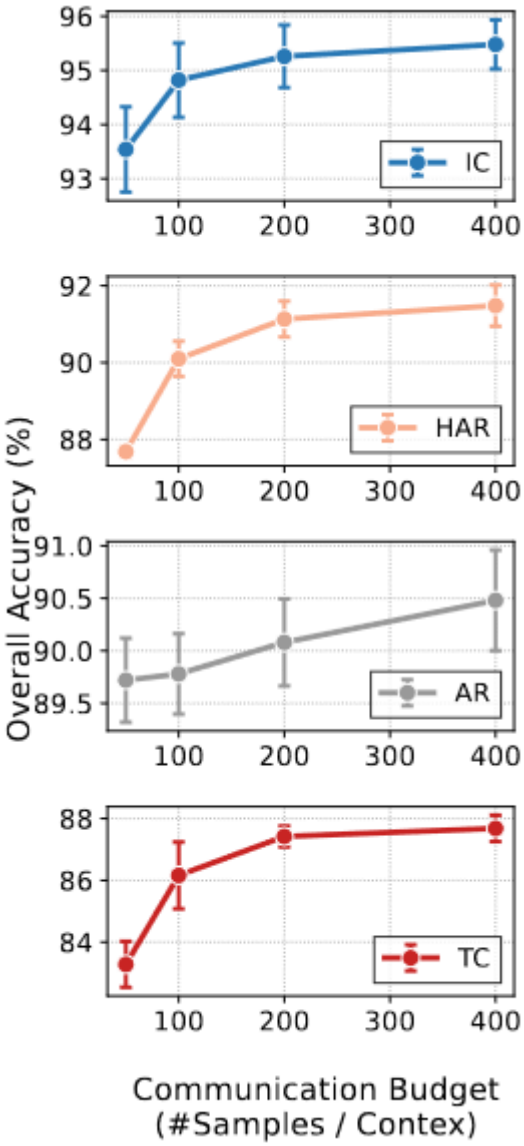
为了性能稳定，我们将每个标签的**聚类数设置为20**。



不同云端数据采样方案的影响

采用不同的抽样方案，包括随机抽样、只针对新情境(α = 0)的最优抽样和考虑所有情境(α = 1)的最优抽样。

对于IC / HAR / AR / TC任务，仅针对新情境的最优数据采样使整体模型精度提高了5.3 / 0.9 / 1.0 / 5.7 %。考虑过去的情景使正确率进一步提升0.9 / 3.9 / 1.5 / 1.7 %。



云端不同通信预算的影响

当每个上下文的丰富数据大小从50增加到100时，Delta的性能显著提高，然后随着数据大小的增加而趋于稳定。

这种鲁棒性突出了Delta对具有不同网络条件的真实世界设备的适用性。

讨论

隐私考虑

在Delta框架中，设备上传的信息包括目录权重，它排除了任何原始用户数据。

权重仅会表示用户数据与目录数据集的相似度，揭示了粗糙的上下文信息，使得原始数据的恢复或识别更具挑战性。为了进一步增强隐私性，可以将安全多方计算、同态加密等安全聚合技术集成到Delta中的通信和计算过程中。

与FL的对比

Delta框架与联邦学习的不同直觉：

- **联邦学习（FL）**的目标是利用设备端数据开发一个能够在不同用户上下文中具有良好泛化能力的全球模型，即进行全球知识的聚合。
- **Delta框架**则侧重于利用云端数据增强本地模型的个性化，针对每个用户的上下文进行本地知识的增强。

联邦学习的局限性：

- 联邦学习的适用性主要受到设备端约束的限制，包括设备数量庞大、高参与率、跨设备数据的异质性以及对通信开销的容忍度。

Delta框架的优势：

- Delta框架将限制转移到云端，假设云服务器可以收集丰富的公共数据，以适应不同用户的需求。这与近年来基于多样化数据集训练大规模模型的成功经验一致。
- 即使面对极为罕见的用户上下文，Delta仍然能够从云端数据中识别最有帮助和相关的数据子集，为现有模型或基于算法的增强方法提供数据支持。

FL与Delta的适用场景：

- 联邦学习和Delta框架适用于不同的场景，二者可以是互补的。